Diffusion and Brownian Motion

Nicolás A. N. Salas

October 16, 2022

Contents

1 Diffusion

Diffusion is usually understood through Fick's law e.g if a free a lotion, there is a current that goes from high density regions to the low density ones, when one mixes this with mass conservation one obtains:

$$\frac{\partial \rho}{\partial t} = D\nabla^2 \rho$$

where D is the diffusion constant, this equation is called difussion equation.

1.1 Important Aspects:

1. For any intial distribution, the standard deviation grows linearly with time.

2. This process is really different from moving in vacuum, there the propagation of the substante can happen *MUCH* faster.

3. Particles of the substante being propagated collide constantly with those of the medium, this is where brownian motion enters the picture.

1.2 Brownian Motion

How do you model this thing? Einstein/Smoluchowski propose to use a random walk to model the random collisions with the particles.

1.2.1 Solution to the Difussion Equation

Lets say we know at the initial condition exactly where our particle is i.e. we got its probability distribution as a Dirac-delta, the solution to this is that after a time t the distribution becomes a Gaussian:

$$\frac{1}{\sqrt{2}}\exp(-\frac{1}{2}(\frac{x}{\sqrt{2}}Dt))$$

whose standard deviation grows as time progresses. This means that one can model Brownian motion by using a random generator that samples a gaussian.

1.2.2 How to Actually Model This: Brownian Dynamics and the Langevin Equation

The idea is to instead of modeling all the molecules in the medium, one limits it to random collisions, the former is called *explicit water* and the former *implicit water*. This is described by the Langevin equation:

$$m_i\frac{dv_i}{dt} = -m_i\gamma_i v_i + F_R(t) + f_i(t)$$

The first term correspond to viscosity, the second one to a random form and the last one the external force.

1.2.3 Van Gunsteren and Berensden Algorithm

The Langevin equation is integrated to obtain the position at $\Delta t + t_n$ as a function of the position at t_n, this method allows to appoximate the equation with fourth order steps and a total of third order, this corresponds to a Verlet method and in fact it becomes the Verlet method when the viscosity goes to zero. The diffusion regime offers another simplification (the exact formulae are long so I am not putting them here). The steps to use this are:

1. Assume $x(t_n)$ and $f(t_{n-1})$ are known

2. Evaluate $f(t_n)$
3. Calculate the derivative of $f(t_n)$ respect to time.
4. Obtain the random value $X_n(\Delta t)$ from a Gaussian distribution.
5. Calculate the position $x(t_{n+1})$

What works for translation can be extened by analogy to rotations.

1.2.4 Random Walk Model for 1D

Any stochastic process (i.e. anything that gives the movement of the obstacle randomly) can produce a diffusion provided the conditions are the same for all points in space and the same at each time step. The idea is the following:

1. The Master Equation The probability of measuring the particle to be at x at time $t + \Delta t$ is

$$P(x, t + \Delta t) = \int_{\infty}^{\infty} dl P(x - l)T(l)$$

i.e. we measure all the possible ways in which a particle can arrive at the point x, where $T(l)$ is the probability of the particle moving l (a probability distribution for the displacements), this is the *Master Equation*. After a taylor expansion and assuming $< T >= 0$ it can be converted into:

$$\frac{P(x, t + \Delta t) - P(x, t)}{\Delta t} = \frac{a^2}{2\Delta t}\frac{\partial^2 P}{\partial x^2}$$

where a^2 is the variance of the distribution T, in the small time step limit this gives the *Fokker-Planck Equation*

$$\frac{\partial P}{\partial t} = \frac{a^2}{2\Delta t}\frac{\partial^2 P}{\partial x^2}$$

if we identify $P(x) \propto \rho(x)$ then the Diffusion equation is recovered:

$$\frac{\partial P}{\partial t} = D\frac{\partial^2 P}{\partial x^2}$$

$$D = \lim_{\Delta t \to 0} \frac{a^2}{2\Delta t}$$

1.2.5 2D Lattice-Gas Diffusion

We will consider a model for the 2D difussion:

1. The space is discretized into cells and each one can have multiple particles with different velocities each. In the implementation the cell stores the velocities of the particles in it.

2. At each time step each cell tosses a coin and based on the result changes the velocities inside it , with probability:

   - p_0 the velocities are not changed
   - p they rotate by 90 degrees
   - p they rotate by 270 degrees, this way there is a symmetry for rotating 90 degrees clockwise or anticlockwise
   - $1 - 2p - p_0$ they rotate by 180 degrees

3. After the coin flip, the particles are moved according to ther velocities.

4. Back to 1.

This produces in the limit of continous time and space a diffusion equation with diffucion constant

$$D = \frac{p + p_0}{2(1 - p - p_0)}$$

1.2.6 1D Diffusion Automata

The idea here is pretty much the same as in the 2D case, but somewhat simpler

1. each cell has two velocity vectors that point to the neighboring cells.

2. Binary Variables: each velocity vector might or might not have an associated particle.

3. Evolution Rule:

   - with probability p the velocities are not changes
   - with probability $1 - p$ the velocities are rotated by 180 degrees
   - after the coin flip and update, the particles move and we go back to step 1.